Module 06

01

Download Coin.java, CoinExample.java, and RandTool.java. Then compile and execute.

Result:

```
Flip #0: 0
Flip #1: 1
Flip #2: 0
Flip #3: 0
Flip #4: 1
```

02

Download StrangeCoin.class. This class models a biased coin in which Pr[heads] is not 0.5. Write a program to estimate Pr[heads] for this coin.

See file `EstimateStrangeCoin.java`.

Result:

```
Pr[heads] = 0.6064
```

03

Modify the above program, writing your code in CoinExample3.java, to estimate the probability of obtaining exactly 2 heads in 3 flips. Can you reason about the theoretically correct answer?

See file `CoinExample3.java`.

Result:

```
Pr[Exactly 2 h in 3 flips]=0.432851   theory=0.432
```

Theory:

$$P_{head} = 0.6$$
$$P = P_{head}P_{head}(1 - P_{head}) + P_{head}(1 - P_{head})P_{head} + (1 - P_{head})P_{head}P_{head}$$
$$= 3P_{head}P_{head}(1 - P_{head})$$
$$= 0.432$$

04

> This exercise has to do with obtaining tails in the first two flips using the biased (Pr[heads]=0.6):
>
> - What is the probability that, in 3 flips, the third and only the third flip results in heads?
> - What is the probability that, with an unlimited number of flips, you need at least three flips to see heads for the first time?
> - Why are these two probabilities different?
>
> Modify the above example, writing your code in CoinExample4.java and CoinExample5.java.

$$P_{head} = 0.6$$

1

Theory:

$$P = (1 - P_{head})(1 - P_{head})P_{head} = 0.096$$

See file `CoinExample4.java`.

Result:

```
Pr[only 3rd is heads]=0.095789  theory=0.096
```

2

Theory:

The complementation of event P "at least three flips to see heads" is event $\overline{P}$ "two flips shows head".

$$P = 1 - \overline{P} = 1 - [P_{head} + (1 - P_{head})P_{head}] = 0.16$$

See file `CoinExample5.java`.

Result:

```
Pr[need at least 3 flips]=0.159836  theory=0.16
```

There are quite difference based on the discription and calculation.

05

> Roll a *pair* of dice (or roll one die twice). What is the probability that the first outcome is odd AND the second outcome is even? Write code in DieRollExample2.java to simulate this event. You will need Die.java. See if you can reason about this theoretically.

See file `DieRollExample2.java`.

Result:

```
Pr[odd+even]=0.249843   theory=0.25
```

Theory:

$$P_{face} = \frac{1}{6}$$
$$P = (P_{face} + P_{face} + P_{face})(P_{face} + P_{face} + P_{face}) = 0.25$$

06

Estimate the following probabilities when two cards are drawn without replacement.

- The probability that the first card is a club *given* that the second is a club.
- The probability that the first card is a diamond *given* that the second is a club.

That is, we are interested in those outcomes when the second card is a club. Write your code for the two problems in CardExample2.java and CardExample3.java. You will need CardDeck.java.

1

See file `CardExample2.java`.

Result:

```
Pr[c1=club given c2=club]=0.23565729864090126   theory=0.23529411764705882
```

2

See file `CardExample3.java`.

Result:

```
Pr[c1=diamond given c2=club]=0.24880134249640404   theory=0.2549019607843137
```

07

List all possible events of the sample space $\Omega = \{heads, tails\}$.

event $heads$

event $tails$

08

How many possible events are there for a sample space of size $|\Omega| = n$?

09

> How many such numbers (probabilities) need to specified for a die roll? How many are needed for a roll of two dice?

There are 6 probabilities need to specified for a die roll.

$$P_1 = \frac{1}{6}$$
$$P_2 = \frac{1}{6}$$
$$P_3 = \frac{1}{6}$$
$$P_4 = \frac{1}{6}$$
$$P_5 = \frac{1}{6}$$
$$P_6 = \frac{1}{6}$$

For 2 die rolls, such number is $6 \times 6 = 36$.

10

> Use the axioms to show that $Pr[\overline{A}] = 1 - Pr[A]$ for any event A.

$$Pr[\overline{A}] = Pr[\Omega - A] = Pr[\Omega] - Pr[A] = 1 - Pr[A]$$

11

> Suppose we draw two cards without replacement:
>
> - What is a compact way of representing the sample space for two card drawings without replacement? What is the size of the sample space?
> - What subset corresponds to the event "both are spades"?
> - What is the cardinality of this subset?
> - What, therefore, is the probability that "both are spades"?
> - What is the probability of "first card is a spade"? Work this out by describing the subset and its cardinality.

1

$$\Omega = \{(x, y) | 0 \le x \le 51, 0 \le y \le 51, x \ne y\}$$

The size of the sample space is $52 \times 52 - 52 = 2652$.

2

$$E_1 = \{(x, y) | 0 \leq x \leq 12, 0 \leq y \leq 12, x \neq y\}$$

3

The cardinality of such subset is $13 \times 13 - 13 = 132$.

4

$$P[E_1] = \frac{132}{2652} = 0.0498$$

5

$$E_2 = \{(x, y) | 0 \leq x \leq 12, 0 \leq y \leq 51, x \neq y\}$$

The cardinality of such subset is $13 \times 52 - 13 = 663$.

$$P[E_2] = \frac{663}{2652} = 0.25$$

12

> Consider this experiment: flip a coin repeatedly until you get heads. What is the sample space?

The sample space is infinity, since the flipping times is not a constant.

13

> Consider an experiment with three coin flips. What is the probability that 3 heads occurred given that at least one occurred?

Let event A to be "3 heads occurred" and event B to be "at least one occurred".

Since event B must happen when event A happens, $P[B|A] = 1$.

The result is $P[A|B]$.

$$P_{head} = 0.6$$
$$P[A] = P_{head} P_{head} P_{head} = 0.216$$
$$P[B] = 1 - (1 - P_{head})(1 - P_{head})(1 - P_{head}) = 0.936$$
$$P[A|B] = \frac{P[AB]}{P[B]} = \frac{P[A]P[B|A]}{P[B]} \approx 0.231$$

14

> Consider two card drawings without replacement. What is the probability that the second card is a club given that the first is a club?

Let event A to be "the first is a club" and event B to be "the second card is a club".

Event AB is equivalent to "get two cards and both cards are club". Based on this, $P[AB] = \frac{C_{13}^2}{C_{52}^2} \approx 0.059$.

The result is $P[B|A]$.

$$P[A] = \frac{13}{52} = 0.25$$

$$P[B|A] = \frac{P[AB]}{P[A]} = 0.236$$

15

> Why? Explain by describing the events C_1 and C_2 as subsets of the sample space and evaluating their cardinalities (sizes).

$$\Omega = \{(x, y)|0 \leq x \leq 51, 0 \leq y \leq 51, x \neq y\}$$
$$C_1 = \{(x, y)|39 \leq x \leq 51, 0 \leq y \leq 51, x \neq y\}$$
$$C_2 = \{(x, y)|0 \leq x \leq 51, 39 \leq y \leq 51, x \neq y\}$$

The size of the sample space is $52 \times 52 - 52 = 2652$.

The cardinality of C_1 is $13 \times 52 - 13 = 663$, and $P[C_1] = \frac{663}{2652} = 0.25$.

The cardinality of C_2 is $52 \times 13 - 13 = 663$ and $P[C_1] = \frac{663}{2652} = 0.25$.

16

> What is $Pr[D_1]$? Compute $Pr[D_1|C_2]$.

Event $D_1 C_2$ is equivalent to "get two cards, the first one is diamond and the second one is club". Based on this, $P[D_1 C_2] = \frac{C_{13}^1 C_{13}^1}{C_{52}^2} \approx 0.064$.

$$P[C_2] = \frac{13}{51} \approx 0.255$$

$$P[D_1|C_2] = \frac{P[D_1 C_2]}{P[C_2]} \approx 0.251$$

17

> Finish (by hand) the calculation above.

$$P[A] = P[A|B]P[B] + P[A|\overline{B}]P[\overline{B}]$$
$$= \frac{3}{51} \times \frac{4}{52} + \frac{4}{51} \times \frac{48}{52} = \frac{1}{13} \approx 0.077$$

18

5% of the population is infected.

$$P[S] = 0.05$$

The probability that the test works for an infected person is 0.99.

$$P[T|S] = 0.99$$

The probability of a false positive is 3%.

$$P[T|\overline{S}] = 0.03$$

Probability that if a test is positive, the person is infected: $P[S|T]$.

Probability that if a test is positive, the person is well: $P[\overline{S}|T]$.

$$P[TS] = P[S]P[T|S] = 0.0495$$
$$P[\overline{S}] = 1 - P[S] = 0.95$$
$$P[T\overline{S}] = P[\overline{S}]P[T|\overline{S}] = 0.0285$$
$$P[T] = P[TS] + P[T\overline{S}] = 0.078$$
$$P[S|T] = \frac{P[TS]}{P[T]} = 0.635$$
$$P[\overline{S}|T] = \frac{P[T\overline{S}]}{P[T]} = 0.365$$

See file `LabTestExample.java`.

Result:

```
Pr[infected given positive]=0.6340914353340709   theory=0.635
Pr[well given positive]=0.36590856466592914   theory=0.365
```

19

- Is 1.0, 1.2, 1.5 an event?
- Is the interval $[1.0, 3.5]$ an event?

$Pr[A > 1]$ means the probability of "waiting time more than 1 hour".

```
Pr[A > 1.0]  = 0.367855
```

$Pr[A > 0.5]$ means the probability of "waiting time more than 0.5 hour".

```
Pr[A > 0.5] = 0.606511
```

An example event is $A > 4$.

1.0, 1.2, 1.5 is not an event.

The interval $[1.0, 3.5]$ is an event.

20

Change the type of interarrival by replacing `true` with `false` above, and estimate both $Pr[first - interarrival > 0.5]$ and $Pr[first - interarrival > 1]$. Note: using `true` results in using *exponential* interarrivals and using `false` results in using *uniform* interarrivals. We'll understand later what these mean.

Change `BusStopExample.java` to:

```java
// BusStopExample.java
//
// Author: Rahul Simha
// Feb, 2008
//
// Estimate Pr[A>1] where A=interarrival time


public class BusStopExample {

    public static void main(String[] argv) {
        double numTrials = 1000000;
        double numSuccesses = 0;
        double value = 0.5;
        for (int n = 0; n < numTrials; n++) {
            BusStop busStop = new BusStop(false);            // True => first
type of distribution.
            busStop.nextBus();
            double interarrival = busStop.getInterarrivalTime();
            if (interarrival > value) {
```

```
            numSuccesses++;
        }


    }
    double prob = numSuccesses / numTrials;
    System.out.println("Pr[A > " + value + "] = " + prob);
    }


}
```

Result when `value = 1`:

```
Pr[A > 1.0] = 0.500319
```

Result when `value = 0.5`:

```
Pr[A > 0.5] = 0.750251
```

21

> Modify the code to estimate the following conditional probability. Let A denote the interarrival time. Estimate $Pr[A > 1.0|A > 0.5]$ for each of the two types of interarrivals (exponential and uniform). What is strange about the result you get for the exponential type? Hint: compare this estimate to that in the previous exercise; that is, compare $Pr[A > 1.0|A > 0.5]$ with $Pr[A > 0.5]$. What is the implication?

See file `BusStopExample.java`.

Result when `isExp = true`:

```
Pr[A > 1 | A > 0.5] = 0.6065100220770934
```

Result when `isExp = false`:

```
Pr[A > 1 | A > 0.5] = 0.6668688212344935
```

It seems that $Pr[A > 1] \neq Pr[A > 1|A > 0.5]$.

$Pr[A > 1]$ means the probability of "waiting time more than 1 hour". $Pr[A > 1|A > 0.5]$ means the probability of "waiting for more than extra 0.5 hour after already waited for 0.5 hour".

22

The average interarrival time for exponential situation may be 0.5.

The average interarrival time for uniform situation may be 1.

23

> Why doesn't the following code work?

```
...
double numBuses = 0;
while (arrivalTime < myArrivalTime) {
 busStop.nextBus();
 arrivalTime = busStop.getArrivalTime();
 numBuses++;
}
...
```

If `arrivalTime > 10`, `numBus` will still be added which is wrong.

24

> Modify the above program to estimate the probability that at least 5 buses have gone by.

Modify `BusStopExample2.java` as following:

```java
public class BusStopExample2 {

    public static void main(String[] argv) {
        double trail = 10000;
        double success = 0;
        for (int i = 0; i < trail; i++) {
            double myArrivalTime = 10;
            BusStop busStop = new BusStop(true);
            double arrivalTime = 0;
            double numBuses = -1;
            while (arrivalTime < myArrivalTime) {
                numBuses++;
                busStop.nextBus();
                arrivalTime = busStop.getArrivalTime();
            }
            if (numBuses >= 5) {
                success += 1;
            }
        }
```

```
        System.out.println("Pr[at least 5 buses have gone] = " + success /
trail);
    }


}
```

Result:

```
Pr[at least 5 buses have gone] = 0.9701
```

25

> Consider the waiting time: the time from your arrival (at 10.0) to the time the next bus arrives.
> Estimate your average waiting time.
>
> - Does this depend on when you arrival (at 10.0 vs. 20.0, for example)?
> - Try this for each of the two types of interarrivals (exponential and uniform).
> - What do you observe is unusual about this wait time?

Yes, it depends on when you arrival.

Modify `BusStopExample2.java` as following:

```java
public class BusStopExample2 {

    public static void main(String[] argv) {
        double trail = 10000;
        double nextArrive = 0;
        for (int i = 0; i < trail; i++) {
            double myArrivalTime = 15;
            BusStop busStop = new BusStop(false);
            double arrivalTime = 0;
            double numBuses = -1;
            while (arrivalTime < myArrivalTime) {
                numBuses++;
                busStop.nextBus();
                arrivalTime = busStop.getArrivalTime();
            }
            nextArrive += arrivalTime;
        }
        System.out.println("Average = " + nextArrive / trail);
    }

}
```

Result for exponential case:

```
Average = 15.977897141796216
```

Result for uniform case:

```
Average = 15.672356116776665
```

It shows that the waiting time is quite close to `arrivalTime`.

26

> Instead of [Coin.java](#) in the example above, use [BizarreCoin.java](#) and see what you get.

Result:

```
Pr[c2=1]=0.499735   Pr[c2=1|c1=1]=0.49881043300238714
```

27

> For the bus-stop problem, define the following events:
>
> - Event B_1 = exactly one bus arrives in the period $[0, 0.5]$.
> - Event B_2 = exactly one bus arrives in the period $[0.5, 1]$.
>
> First, without writing a program, what does your intuition tell you about whether these events are independent? Next, write a program to estimate $Pr[B_2]$ and $Pr[B_2|B_1]$. Do this for each of the exponential and uniform cases.

It seems that they are independent.

In uniform case, $Pr[B_2] \approx 0.250$, $Pr[B_2|B_1] \approx 0.249$.

In exponential case $Pr[B_2] \approx 0.239$, $Pr[B_2|B_1] \approx 0.302$.

28

> Three subcontractors - let's call them A, B, C - to a popular clothing store contribute roughly 20%, 30% and 50% of the products in the store. Turns out that 6% of A's products have some defect; the defect numbers for B and C are 7% and 8% respectively. A product is selected at random and found to be defective. What is the probability that it came from subcontractor A?

Based on the description (let event D to be "defective"):

$$P[A] = 0.2$$
$$P[B] = 0.3$$
$$P[C] = 0.5$$
$$P[D|A] = 0.06$$
$$P[D|B] = 0.07$$
$$P[D|C] = 0.08$$

Probability of "defective from subcontractor A" is $P[A|D]$.

$$P[A] + P[B] + P[C] = 1 \implies P[D] = P[D|A]P[A] + P[D|B]P[B] + P[D|C]P[C] = 0.073$$
$$P[A|D] = \frac{P[AD]}{P[D]} = \frac{P[D|A]P[A]}{P[D]} \approx 0.164$$

29

> We have seen four key examples: a coin, a die, a deck of cards, and the bus-stop. Examine the code in each of: Coin.java, Die.java, CardDeck, and BusStop.java. See if you can make sense of how random outcomes are created. (There's no written answer required for this question.)

(no written answer)